

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-044093
(43)Date of publication of application : 14.02.2003

(51)Int.Cl.

G10L 15/28
G10L 15/00
G10L 15/22

(21)Application number : 2002-132052 (71)Applicant : MICROSOFT CORP
(22)Date of filing : 07.05.2002 (72)Inventor : WANG KUANSAN
HON HSIANG-WUEN

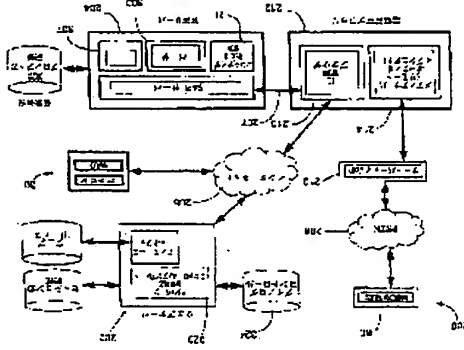
(30)Priority

Priority number : 2001 289041 Priority date : 04.05.2001 Priority country : US
2001 960229 20.09.2001 US
2002 117141 05.04.2002 US

(54) METHOD FOR SERVER FOR WEB ENABLED SPEECH RECOGNITION AND RECORDING MEDIUM

(57)Abstract:

PROBLEM TO BE SOLVED: To enable a server for web enabled speech recognition, which is used for providing speech recognition in the server/client architecture of the Internet or the like, to have a unified architecture. SOLUTION: A markup language for execution on a client device in a client/ server system contains an instruction for unifying at least one of events related to recognition, a GUI event and a telephony event on a non-display, voice input based client device 80 and a multimodal based client 30 for a web server 202 to interact with each of client devices. A recognition server 204 is provided as well for receiving data showing input data provided to the client device and an instruction of a grammar to be used for recognition.



LEGAL STATUS

[Date of request for examination] 06.05.2005
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]

(19)日本国特許 (JP) (12) 公開特許公報 (A)

(11)特許出願公開番号
特開2003-44093
(P2003-44093A)

(43)公開日 平成15年2月14日(2003.2.14)

(51)Int.Cl. FI
G10L 15/28 G10L 3/00
15/00 571Z 5D015
15/22 551A
551P
561H
571U

審査請求 未請求 請求項の数20 OL (全 49 頁) 最終頁に続く

(21)出願番号 特開2002-132052(P2002-132052)
(22)出願日 平成14年5月7日(2002.5.7)
(31)優先権主張番号 60/289,041
(32)優先日 平成13年5月4日(2001.5.4)
(33)優先権主張国 米国 (US)
(31)優先権主張番号 09/960,229
(32)優先日 平成13年9月20日(2001.9.20)
(33)優先権主張国 米国 (US)
(31)優先権主張番号 10/117,141
(32)優先日 平成14年4月5日(2002.4.5)
(33)優先権主張国 米国 (US)

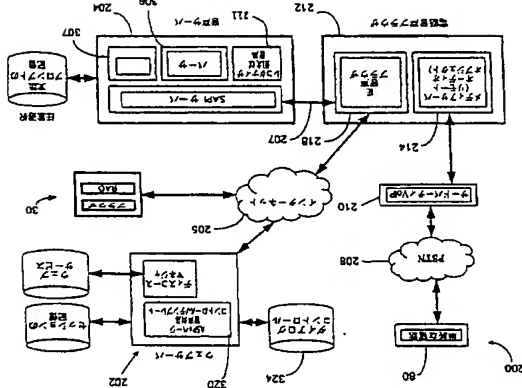
(71)出願人 301055633
マイクロソフト コーポレーション
MICROSOFT CORPORATI
ON
アメリカ合衆国 ワシントン州 98052-
6399 レッドモンド ワン マイクロソフ
ト ウェイ (番地なし)
(74)代理人 100077481
弁護士 谷 穂一 (外2名)

(54)【発明の名称】 ウェブ対応音声認識用サーバの方法および記録媒体

(57)【要約】

【課題】 インターネットなどのサーバ/クライアントアーキテクチャで音声認識を提供するのに使用されるウェブ対応音声認識用サーバに、統一したアーキテクチャを構建させる。

【解決手段】 クライアント/サーバシステムのクライアントデバイスで実行するためのマークアップ言語は、非表示式、音入カベースのクライアントデバイス80と、各クライアントデバイスと対話するウェブサーバ202用のマルチモーダルベースのクライアント30とにおける、認識に際してのイベント、GUIイベント、および電話イベントのうち少なくとも1つを、各クライアントデバイスと対話するウェブサーバ202のために統一する命令を含む。クライアントデバイスに提供された入力データを示すデータと、認識に使用する文法の指示とを受信する認識サーバ204も提供される。



インデックスを日常活動で使用する頻度は増す一方である。現在では、こうしたデバイスを作動させるのに使用されるマイクプロセッサに利用できる処理能力が増大したことにより、これらデバイスの機能性が高まっており、場合によっては機能を一体化している。例えば現在、携帯電話の多くは、アドレス、電話番号などの個人情報やアドレスを記憶して使用できるだけでなく、インターネットのアクセスおよびブラウザにも使用することができる。

【0003】こうしたコンピュータインデックスをインデックスネットワークに使用し、あるいは他のサーバ/クライアントアーキテクチャで使用するところから、情報をコンピュータインデックスに入力することが必要となる。不都合なのは、移行を容易にするためにこうしたデバイスを可能な限り小さくしたいという要求があり、利用可能なコンピュータインデックスデータの表面積が限られているために、アルファベットの全文字を個別のボタンとして備える従来のキーボードが通例は不可能であることである。

【0004】最近、VoiceXML (音声拡張可能マークアップ言語) の使用によるなどの音声ブラウザが進歩し、電話だけを使用してインターネットコンテンツにアクセスすることが可能になっている。このアーキテクチャでは、ドキュメントサーバ (例えばウェブサーバ) が、VoiceXMLインタープリタを通じてクライアントからの要求を処理する。ウェブサーバはそれに応じてVoiceXMLドキュメントを生成することができ、このドキュメントはVoiceXMLインタープリタによって処理し、ユーザに対して音声としてレンダリングされる。ユーザは、音声認識を通じて音声コマンドを使用することにより、ウェブをナビゲートすることができ。

【0005】VoiceXMLは、フロー制御タグを用いるマークアップ言語であるが、フロー制御は、イベントインギ (eventing) および個別のスクリプトを含むHTML (ハイパーテキストマークアップ言語) のフロー制御モデルには従わない。VoiceXMLは一般に、電話ベースの音声のみの対話に特に適したフォーム解釈アルゴリズムを含むが、このアルゴリズムでは通例、ユーザから得られる情報をシステムまたはアプリケーションによって制御する。グラフィカルユーザインタフェースも提供し、クライアントサーバ間で利用することのできるアプリケーションにVoiceXMLを直接組み込むには、開発者は、2つの形態のウェブページングを習得する必要がある。すなわち、VoiceXMLのオーサリングと、HTML (など) を使用した一部制モデルであるが、これらはそれぞれ異なるフロー制御モデルに従っている。

【0006】

【発明が解決しようとする課題】したがって、インターネットなどのサーバ/クライアントアーキテクチャで普

【0018】上記の構成式またはモバイル型のコンピュータインデックス以外に、本発明は、一般的なデスクトップコンピュータなど数多くの他のコンピュータインデックスにも使用できることが理解されよう。例えば、身体能力が限られたユーザにとって完全な英数字キーボードなどの従来の入力装置の操作が困難である場合には、本発明は、そのようなユーザがコンピュータまたは他のコンピュータインデックスにテキストを入力することを可能にする。

【0019】本発明はまた、数多くの他の利用または特殊目的のコンピュータインデックス、環境、または構成での動作が可能である。本発明とともに使用するのに適した周知のコンピュータインデックスシステム、環境、および/または構成の例には、従来型の (regular) サバコンピュータ、携帯用デバイスまたはラップトップデバイス、マルチプロセッサシステム、マイクプロセッサベースのシステム、セットトップボックス、ログラム、メインフレームコンピュータ、上記のシステムまたはデバイスなどの任意のものを含む分散型コンピュータインデックス環境が含まれるがこれらに限定するものではない。

【0020】以下は、図4に示す汎用コンピュータ120の簡単な説明である。ただし、この場合もコンピュータ120は、適切なコンピュータインデックス環境の一例に過ぎず、本発明の使用または機能性の範囲に関して何らの制限を示唆するものではない。また、コンピュータ120は、この図に示す構成要素のいずれか、またはその組合せに関する依存性や要件を有するものとも解釈すべきではない。

【0021】本発明は、プログラムモジュールなどコンピュータで実行するコンピュータ実行可能命令の一般的な状況で説明することができる。一般に、プログラムモジュールには、特定タスクを実行する、または特定の抽象データタイプを認識する、ルーチン、プログラム、オブジェクト、コンポーネント、データ構造などが含まれる。本発明はまた、通信ネットワークを通じてリンクした遠隔処理装置によってタスクを実行する分散型コンピュータインデックス環境でも実施することができる。分散型コンピュータインデックス環境では、プログラムモジュールは、メモリ記憶装置を含む、ローカルおよびリモートどちらのコンピュータ記憶媒体に置いてよい。以下で、図面の助けを借りて、プログラムおよびモジュールによって実行するタスクを説明する。当業者は、この説明および図面をプロセッサ実行可能命令として実施することができ、この命令はどの形態のコンピュータ読み取り可能な記憶媒体にも書き込むことができる。

【0022】図4を参照すると、コンピュータ120の構成要素には、プロセッサ140、システムメモリ15

0、およびシステムメモリを含む各種システム構成要素をプロセッサ140に結合するシステムバス141が含まれるが、これらに限定しない。システムバス141は、メモリバスまたはメモリコントローラ、周辺バス、および各種バスアーキテクチャのうち任意のものを使用したローカルバスを含む各種バス構造のうち任意のものでよい。このようなアーキテクチャには、例えば、ISA (Industry Standard Architecture) バス、USB (Universal Serial Bus)、MCA (Micro Channel Architecture) バス、EISA (Enhanced ISA) バス、VESA (Video Electronics Standards Association) ローカルバス、およびPeripheral Component Interconnect) バスが含まれるがこれらに限定するものではない。コンピュータ120は、通例、各種のコンピュータ読み取り可能な記憶媒体を含む。コンピュータ120からアクセス可能な記憶媒体は、コンピュータ120からアクセス可能な任意の任意の記憶媒体、リムーバブルおよび非リムーバブル記憶媒体を含む。例えば、コンピュータ読み取り可能な記憶媒体は、コンピュータに限定しない。コンピュータ記憶媒体には、コンピュータ可読命令、データ構造、プログラムモジュール、または他のデータを記憶するための任意の方法または技術に実施された、揮発性および非揮発性、リムーバブルおよび非リムーバブル記憶媒体が含まれる。コンピュータ記憶媒体には、RAM、ROM、EPROM、フラッシュメモリ、または他のメモリ技術、CD-ROM、デジタル多用ディスク (DVD)、またはその他の光ディスク記憶、磁気カセット、磁気テープ、磁気ディスク記憶または他の磁気記憶装置、あるいは所望の情報の記憶に使用することができ、コンピュータ120からアクセスすることが可能な任意の他の媒体が含まれるがこれらに限定するものではない。

【0023】通信媒体は、通例、搬送波または他の搬送機構などの変調データ信号中のコンピュータ可読命令、データ構造、プログラムモジュール、または他のデータを搬送し、また任意の情報伝送媒体を含む。用語「変調データ信号」とは、情報を信号中に符号化するような方式で、その特性の1つまたは複数の特徴または変化する特徴を意味する。例として、通信媒体には、配線式ネットワークまたは直接配線接続などの配線媒体、および音響、F R、赤外線、および他の無線媒体などの無線媒体が含まれるが、これらに限定しない。上記の媒体の任意の組合せも、コンピュータ読み取り可能な記憶媒体の範囲に含まれるものとする。

ログラム195をリモートコンピュータ194に常駐するものとして示しているが、これに限定しない。図のネットワーク接続は例示的なものであり、コンピュータ間に通信リンクを確立する他の手段を使用してよいことは理解されよう。

【0030】図5に、本発明で実施することのできるウェブページ形成のアーキテクチャ200を示す。一般に、ウェブサーバ200中に記憶された情報には、モバイルデバイス30（本明細書では、入力の形態に基づき、適宜、表示画面、マクロプロファイル、カメラ、タッチセンシティブパネルなどを有する他形態のコンピュータシステムを指す）を通じて、または情報や音声をよるデータパスをも含む）を通じて、または情報や音声を伴うように応答して電話機80が生成される。またキーを押下して要求する電話機80を通じてアクセスすることができ、電話機80の場合には、ウェブサーバ200からの情報を、音声のみによりユーザに提供す。

【0031】より重要なのは、情報を変換バイナリ30を生成して送信することである。この変換は、電話線を用いて電話機80を通じて得られるか、または無線で、単一の接続サーバ200が80を通じて得られるかのいずれかに関係なく、同一の接続サーバ200が行うことができる。また、動作モードもサポートすることができる点でアーキテクチャ200が統一されていることである。さらに、アーキテクチャ200は、周知のマークアップ言語（例えばHTML、XHTML、cHTML、XML、WMLなど）の拡張を使用して動作する。したがって、ウェブサーバ202に記憶された情報には、これらのマークアップ言語で使用される周知のGUI方式を用いてアクセスすることもできる。周知のマークアップ言語の拡張を使用することで、ウェブサーバ202でのオペレーティングが容易になり、現在存在するレガシーアプリケーションも、音声認識を必要としないように修正することができ、ユーザが音声認識を利用しやすくなる。

【0032】一般に、デバイス30は、ウェブサーバ20が提供するHTMLページ、スクリプトなどを実行可能とする。一例として、音声(voice)認識が必要な場合には、デジタル化したオーディオ信号または音声特徴量などの音データ(オーディオ信号または上記のようにデバイス30で前処理する)、音声認識中に使用する文法または言語モデルの指示とともに、認識サーバ204に提供される。認識サーバ204の実施態様は多くの形態を有し、それが可能であり、そのうちの1つを図示したが、一般にはレコグナイザ211を含む。認識の結果は、所望の場合、または適切な場合にはローカルのレンダリングのためにデバイス30に戻される。認識と、使用する場合には任意のグラフィカルユーザインタフェースとを通じて情報と編集すると、必要な場合には、デバイス30はその情報をウェブサーバ202に送信し、そこでさらに処理を行い、さらにHTMLページ/スクリプトを受信する。

【0033】図5に示すように、デバイス30、ウェアサーバ202、および認識サーバ204は其前に（c0

プロジェクトまたは要素を提供するドキュメントオブジェクトモデルに従う。各要素については付録中で詳細に説明するが、一般に、要素には属性、プロパティ、メソッド、イベント、および/または他の「子」要素を含むことができる。

【0043】本明細書で、エクステンションは、ブラウザを実行するデバイスの機能に応じて、異なる2つの「モード」で解釈できることにも留意されたい。第1のモード「オブジェクト」では、金機能を利用することができ、アプリケーションによるエクステンションのプログラム上の操作は、そのデバイスのブラウザが使用可能にするとの機構でも実行することができる。これは、例えば、HTMLブラウザにおける「Scripting」イベント、XMLブラウザにおけるWMLScriptイベント、WMLブラウザにおけるWMLScriptイベントなどの機構などがある。この理由から、エクステンションのコアロジックおよびメンソッドの小セグメントだけを定義すればよく、これらは、デバイスすなわちクライアント側に存在する任意のプログラム機構によって操作される。オブジェクトモードは、イベントのオブジェクトおよびスクリプティングを提供し、またより多くの機能を提供して、データのオーサに、音声対話に対するより細かなクライアント側におけるコントロールを与え、本明細書で使用する場合、ブラウザおよびスクリプティングをサポートするブラウザを「アップレベルブラウザ」と呼ぶ。この形のブラウザは、エクステンションのすべての属性、プロパティ、メンソッド、およびイベントをサポートする。アップレベルブラウザは、通常、より高い処理能力を持つデバイスで使用する。

【0044】エクステンションは、「宣言モード」でもサポートすることができる。本明細書で使用する場合、宣言モードで動作するブラウザを「ダウンレベルブラウザ」と呼び、これは完全なイベントリングおよびスクリプティング機能はサポートしない。代わりにこの形のブラウザは、所与のエクステンションの宣言的側面（すなわちコア要素および属性）をサポートする。DOM（ドキュメントオブジェクトモデル）オブジェクトのプロパティ、メンソッド、およびイベントのすべてはサポートしない。このモードは専ら宣言構文だけを用い、さらに、SMIL（同期化マルチメディア統合言語）2.0などの宣言マルチメディア同期化および協調機構（同期マークアップ言語）と併せて使用することができる。ダウンレベルブラウザは、通常、処理能力が限られたデバイスで使用する。

【0045】ここで、特定の入力モードについて論じておきたい。詳細には、音声認識を、少なくともディスタントレベールと併せて、そして別の異態形態ではポイントイングデータ入力とも併せて使用して、データ入力フィールドを指定するものと併せて使用。具体的に、このモードのデータ入力では、ユーザは一般に、いくつかのモードを選択

採し、それに対応する情報を提供するかを制御することができ。例えば、図6の例では、ユーザはまずフィールド252にクレジットカード番号を入力し、次いでフィールド250にクレジットカードの種類を入力し、最後にフィールド254に有効期限日を入力することができ

きる。同様に、ユーザは、所望の場合にはフィールド27
52に英字、顔つた音声を入力修正することもできる。下記
で説明するように音声認識と組み合わせた、平易で自
然なナビゲーション形態が提供される。本発明で使用する
場合、フィールドを自由な形で選択することを可能に
する画面表示と、音声認識の両方を使用するこの形の入り
方を「マルチモーダル」と呼ぶ。

【0046】再び図8を参照すると、HTMLマークア
ップ言語のコードが示されている。図7に示すHTML
コードと同様に、このコードも、本体部分270および
スク립ト部分272を含んでいる。また図7に示すコ
ードと同様に、図8に示すユーザは、フォームの位置
および実行するアクションのタイプに関する指示を含む。

ファイル250、252、および254それぞれへ、
20 情報の入力は、各々コード部分280、282、および
284によって制御または実行する。初期にコード部分
280を参照すると、例はデバイス30のスタイラス
33を使用してファイル250を選択すると、イベン
トonClickが開始され、これによりスクリプ
ト部分272の関数「talk」が呼び出されるか、ま
たは実行される。このアクションは、一般にファイ
250に予想されるデータタイプと関連付けられた、音
声認識で使用する文法を起動する。複数の入力技術（例
えば音声とペンクリック/ローラ）を使用するこの種の
30 対話を「マルチモーダル」と呼ぶ。
【0047】図8に例示する音声認識エクステンション
は、クワイアントのプラグザンにおいてデフォルトの視覚
表現を有さないことに留意されている。これは、多くのア
プリケーションでは、オーサザが、アプリケーション仕様
のグラフィック機構をソーススペースで使用するることによ
り、ページの各種コンポーネントの音声使用可能を知ら
せることを想定しているためである。それでも、視覚的な
表現が望ましい場合には、エクステンションをそのよ
うに修正することができる。

【0048】再び文法を参照すると、この文法は、文脈自由文法、N文法、ハイブリッド文法などの情文法で、あるがこれらに限定しない。(言うまでもなく、それに対応する形態の認識を利用する際には、DTMF文法、手書き文法、ジェスチャ文法、および画像文法を使用する。本明細書で使用する場合、「文法」とは認識を行うための情報を含み、別の実施形態では、例えば特定のフィールドに入力されるとことが推定される入力に対応する情報を含む。) マークアップ言語が諸語の最初エクステンションを含む新しいコメントロール290 (本明細書では「resco」と題する)は様々な変義を含むが、その

【0056】認識を中止またはキャンセルすることができ、他の属性には、「bubbleTimeout」属性295があるが、これは285の音声の検出後に認識サーバ204が結果を返さなければならない期間である。この期間を超えると、エラー発生の有無に応じて異なるイベントが発生される。例えば、例外的に音声が高くなる場合、認識サーバ204がなおオーディオの処理を行っている場合は、「onNoRecog」属性293を実行する。しかし他の何らかの理由で「bubbleTimeout」属性295を超えた場合は、認識エラーの可能性が高くなり、「onTimeout」イベント297が発生する。同様に「maxTimeout」属性299も提供することができる。これは、認識の開始283から結果をクライアント30に返すまでの期間である。この期間を超えると、「onTimeout」イベント297が発生される。

【0057】ただし、「endSilence」属性301以上の期間を超えた場合、これは認識が完了していることを示唆するが、この場合は認識サーバ204が自動的に認識を中止し、その結果を返す。認識サーバ204は、信頼度の測定を実施して、認識結果を返すべきかどうかを判定できることに留意されたい。信頼度の測定値が閾値を下回る場合は、「onNoRecog」属性293を発行し、一方信頼度の測定値が閾値を上回る場合は、「onNoRecog」属性303および認識結果を「onNoRecog」属性303から「onNoRecog」属性293に発行する。したがって図14は、「自動モード」で明示的なstop()の呼び出しが行われていない状況を表している。

【0058】図15は、認識サーバ204の「シングルモード」の動作を図式的に表したものである。「自動モード」の関連で上記で説明した属性およびイベントを適用することができ、したがって同じ参照番号で示している。しかし、この動作モードでは、stop()呼び出し305を、スケジューラ281上に示している。stop()呼び出し305は、ユーザによる「ペンアップ」などのイベントに相当する。この動作モードでは、認識結果を返すことは、明示的なstop()呼び出し305によって制御される。すべての動作モードの場合と同じく、「onSilence」イベント291は、「initialTimeout」期間289内に音声が発出されない場合に発生するが、この動作モードでは認識を中止しない。同様に、stop()呼び出し305以前の認識不可能な発声によって生成される「onNoRecog」イベント293によっても認識は中止されない。ただし、「bubbleTimeout」属性295または「maxTimeout」属性299と関連付けられた期間を超えた場合は、認識を中止する。

【0059】図16は、認識サーバ204の「複数モード」の動作を図式的に表している。上記で指摘したよう

に、この動作モードは、「オープンマイクオフオン」またはディクテーションのシナリオで使用。一般に、この動作モードでは、明示的なstop()呼び出し305が受け取られるか、または「bubbleTimeout」属性295または「maxTimeout」属性299に関連付けられた期間を超えたまで、閾値を置いて認識結果を返す。ただし、「onSilence」イベント291、「onNoRecog」イベント303、または「onNoRecog」イベント293のいずれかが発生すると、これらによって認識は中止されないが、「bubbleTimeout」期間および「maxTimeout」期間のタイマがリセットされることに留意されたい。

【0060】一般に、この動作モードでは、stop()呼び出し305が受け取られるまで、認識されるフレーズごとに、「onRecog」イベント303を発行し、結果を返す。認識不可能な発声のために「onSilence」イベント291が発行された場合は、これらのイベントを報告するが、認識は継続する。【0061】上記で触れたように、フィールドに関連付けられた1つまたは複数のrecogオブジェクトを起動するが、これには、少なくとも1つの文法を使用するに付いての指示を認識サーバ204に提供することが含まれる。この情報は、クライアント30で記録して認識サーバ204に送信した音声データを伴うことができる。上記で指摘したように、音声データは、ユーザが入力した音声に関連づけられたストリーミングデータを含むことができ、あるいは音声認識中に使用する音声の特徴を示す、前処理済みの音声データを含むことができる。別の実施形態では、クライアント側の処理に音声データの正規化も含むことができ、認識サーバ204が受け取る音声データが、クライアントごとに比較的に異なるようにする。これにより認識サーバ204の音声処理が簡略化され、認識サーバを、クライアントおよび通信経路のタイプにステータスにすることができ、認識サーバ204のスケラビリティをより容易にすることができ。

【0062】認識サーバ204から認識結果を受け取ると、その認識結果を対応するフィールドと関連付け、必要な場合はクライアント側で確認またはチェックを行うことができる。現在クライアントがレンダリングしているコードと関連付けられたすべてのフィールドを完了すると、アプリケーション処理のためにその情報をウェブサーバ202に送信する。前述の内容から、ウェブサーバ202は、認識に適したコードまたはページ/スクリプトをクライアント30に提供しているが、認識サーバ204によって行われることが明白である。ただし、本発明は、認識サーバ204をウェブサーバ202とまとめて配置する、または認識サーバ204をクライ

(incur) ことなく、そのページのオブジェクトのメソッドを起動することができる。

【0075】上記の実施形態では、「bind」要素は、認識結果をフォーム中またはウェブページ中のフィールドに割り当てるための属性「targetElement」および「targetAttribute」しがある。別の実施形態では、「bind」要素は、オブジェクトメソッドの起動のために加える「targetMethod」も含む。「targetMethod」の使用および機能は、スクリプティングの領域にとって非常に重要な技術である。例えば、次の構文を使用して、オブジェクト「OBJ1」の「X」メソッドを起動することができる。

```
<bind TargetElement = "OBJ1" TargetMethod = "X"
...>
```

ここに示す例はHTML/XHTMLのイベント構文に従っているが、当業者にとつては、<bind>の使用を一般化して、他のイベントリング機構を使用することは平易であることに留意されたい。他のイベントリング機構には、W3Cドキュメントオブジェクトモデルレベル2またはレベル3のイベントリング規格、ECMA共通言語仕様(CLI) イベントモデル、Java (登録商標) プログラミング言語イベントモデル、W3C同期マルチメディア統合言語(SMIL)、および近々登場するW3CのXMLイベント規格提案が含まれるが、これらに限定するものではない。

【0076】図17および18は、クライアント、特にダウンレブルブラウザで実行可能なマークアップ言語のページである。この例では、音声プロンプトを通じてユーザに希望する飲料を尋ねている。このシステムは次いで、どの飲料が注文されたかを確認する。認識結果に応じて、「bind」要素は、宣言した論理を使用して実行を増く。飲料を確認すると、そのフォームをウェブサーバ202に再度提出するが、これらにスクリプティングは一切用いない。

【0077】一般に、図17および18のマークアップ例は、データ部分350、音声部分352、およびユーザインタフェース部分354、356、および358を含む。部分354は、全般的な質問から、ユーザが希望する飲料についての認識結果を受け取り、初稿認識フローを誘導して、クリームや砂糖が必要かどうかについて再度指示を促し、尋ねるか、または注文された飲料を確認する。詳細には、部分356は、クリームや砂糖は注文された場合にはその認識結果を受け取る。部分358は、飲料の確認についての認識結果を受け取る。部分360は、新しいメッセージングオブジェクト「SMEX」を用いる制御部分である。「SMEX」については下記でさらに説明する。

【0078】上記で指摘したように、本発明のこの態様の「bind」要素はオブジェクトメソッドの起動を含

「expiry_date」と関連付けられた情報を含む。この例では、電話音声ブラウザ212は、「onRecog」0として認識済みの音声を受け取る。電話機80から受け取った音声データと、「do_field」関数文法の使用の指示を認識サーバ204に送信し、関数「handle」が呼び出され、または実行されるが、これには音声データから認識されたフィールドの一部またはすべての値を関連付けることが含まれる。すなわち、認識サーバ204から得る結果は、各フィールドについての指示も含んでいる。この情報は構文解析し、405で指定されるバインド規則に従って対応するフィールドと関連付ける。図5に示すように、認識サーバ204はパラ309を含むことができる。

【0071】図7、8、9、10、11、および12から、非常に類似したウェブ開発フレームワークを使用する。データの提示も、これらの各場合で非常に類似している。さらに、データ提示とフロー制御を分離することにより、異なるアプリケーション(システム主導型と組合型)間、または異なるモダリティ間(GUIウェブベース、音声のみ、およびマルチモーダル)での再使用性を最大限にすることができる。また、これにより、電話機がディスプレイおよびディスプレイ30と同様の機能を含む場合に、音声のみの動作から電話、そしてマルチモーダル動作への自然な拡張が可能になる。付録Aで、以上で説明したコンローラおよびオブジェクトの詳細にさらに提供する。

【0072】図17および18は、クライアント、特にダウンレブルブラウザで実行可能なマークアップ言語のページである。この例では、音声プロンプトを通じてユーザに希望する飲料を尋ねている。このシステムは次いで、どの飲料が注文されたかを確認する。認識結果に応じて、「bind」要素は、宣言した論理を使用して実行を増く。飲料を確認すると、そのフォームをウェブサーバ202に再度提出するが、これらにスクリプティングは一切用いない。

【0073】スクリプティングは有用でありうるが、多くの者は、例えばセキュリティ問題から必ずしも最も良いブラウザが実装形態であるとは限らないと見ている。したがって、本発明のさらに別の実施形態または態様では、「bind」要素は「recog」同様の) 高レベル要素であり、他のより豊富なプロパティとともに提供され、実際、それ自体ではスクリプティングを用いずにスクリプティングを実際に構築することができる。

【0074】スクリプティングを用いない場合、あるいは下記で述べる本発明の態様を使用しない場合、高度なダイアログ効果など下記で述べる機能の一部は、ページを再度ウェブサーバ202に提出し、そこでアプリケーションロジックを実行して新しいページを生成し、そのページを再びクライアントデバイスに送信することによってのみ実現することができる。本発明のこの態様により、プログラマは、サーバへのラウンドトリップを招く

オブジェクト (ここでは「smex」と書く。例えばブラウザインタプリタで実行するアプリケーションからの) にも適用できることに留意されたい。図 17 および 18 の例では、クライアントデバイスで実行される電話

アプリケーションが呼を出すと、このページが実行
【0089】再び図5を参照すると、ウェブサーバ20
2は、サーバ側のプラグイン宣言オーサリングツールす
される。部分360で、「bind」要素は、メッセー

「welcome」プロンプトを再生し、
「reco_drink」オブジェクトを実行すること
し、Call_connected」を受け取る。
なわちモジュール320を呼びこむことである（例えば、
マイクロソフト社によるASPまたはASP+、あるいはJSPなど）。サーバ側のプラグインモジュール32

により認識を開始する。音声サーバ204から受け取る100は、クライアント側のマークアップと、さらにウェブブラウザ202にアクセスするクライアントのタイプについて固有形値のマークアップも動的に生成することが可能である。メッセージの一例は、希望のプログラム

できる。クライアント情報は、クライアント/サーバ間
係が保たれに確立されたときにウェブサーバ202に提供
される。クライアント201の情報は、クライアント201
からウェブサーバ202に提供される。

例えば、これにより、キーボードから入力するテキストの自然言語処理バーサスのようにマークアップを使用できる。この方式で、サーバ側のプラグインモジュールを検出するモジュールまたはルーチンを含むことができる。この方式で、サーバ側のプラグインモジュールを検出するモジュールまたはルーチンを含むことができる。この方式で、サーバ側のプラグインモジュールを検出するモジュールまたはルーチンを含むことができる。

るためのプロパティを含んでいる。同様に、プロンプト
うになる。付録Aのrec o要素は、この機能を実行す
ル320は、それぞれの音声認識シナリオ、すなわち電
話機80を通じて音声のみ、あるいはマルチモーダル型
のデバイス30に対する、クライアント側のマークアップ
20

nervertex]を使用することにより、動的コンテンツまたはオーディオウェーブファイル用のテキストメ

シメーンを促進することができる。イベンティングは、多数の異なるクライアントのアプリケーション・サードパーティが大幅に容易になる。

【0090】クライアント側マークアップの動的な生成に
加えて、図8、9および10のマークアップ例を用い
た、図6に示すようなクレジットカード番号の入平など
が、ブラウザで利用できるメッセージを有する際に送ら
れるクライアントデバイスで実行するアプリケーション

【0087】このように、「smex」すなわちメッセ 30 の高レベルのダイアログモジュールは、アプリケーション・ソフトウェアの開発者が使用するために、記憶装置 31

ンポーネントにより、ここに述べるようなマージンオブシエリットにより、ここに述べられる他のコ
ツプタグを、クライアントデバイスで実行される他のコ
ンポーネントまたはアプリケーションに拡張することが
できる。一般に、高レベルダイアログモジュール3
24に記述するクライアント側コントロールとして実装するこ
とができる。一般に、高レベルダイアログモジュール3
24は、開発者が指定するパラメータに基づいて、音声

可能になる。別の例として、このメタセージオブジェクトを使用して、クライアントデバイスで実行される感覚ト側のみおよびマルチモーダルの両シナリオで、クライアント側のマークアップおよびスクリプトを動的に生成する。高レベルダイアログモジュールは、開発者のニーズ

る。TTYコンポーネントは、音声認識を使用するのでなく、ユーザが入力した内容のメッセージを提供し、ユーザの要求に応じて、クニオアノミのマークアップを生成するに適合するクライアント側のマークアップを生成することができる。例えば、クレジットのパラメータを含むことができる。

ら受け取った場合と同様に使用する。すなわち、メッセージを構文解析して、フォームのフィールドに割り当て

40 アップスクリプトが許可すべきクレジットカードの種類
を指定するパラメータを含むことができる。サーバー側ブ

るか、あるいは上記の「recol」、「gamma
r」、または「bind」要素を使用して他の処理を行
うことができる。このメッセージまたは「smexl」オ
ペレータは、図13に示すように、ASAP+ベージの
ラゲインモジュール320で使用する。

プロジェクトについては、付録 A でさらに説明する。
【0088】「bind」要素は「for」属性も含む
逸脱せずに、形態および詳細を変更することが可能であ
て説明したが、当業者は、本発明の趣旨および範囲から

ツブ

1 概要

【0092】付録A

ることを理解されよう。

<bind for = "prompt" event = "onComplete" targetE 50 以下のタグは、ドキュメントが音声を入力媒体またはは

は、認識結果を指定するためのセマンティックマークアップ言語 (SML) を含むXMLドキュメントでよい。その内容は、意味値、話された実際の単語、および信頼度スコアを含む。SMLは、代替の認識選択度 (N番目

により認識結果におけるものなど)も含むことができ
 <smI confidence="40">

<travel text= シェトルカ
 <origin_city confider
 </origin_city>

```
</dest_city>
</travel>
```

[0105] 文法中 (in-grammar) 認識は、
</sm>

セマンティックマークアップ言語すなわちSMLでXMLドキュメントを生成することになっているので、SMLドキュメントからバインドする値は、XPathのエク

リを使用して参照する。また、値をバイナリ形式のデータとして保存し、このデータをデータベースに格納する。また、値をバイナリ形式のデータとして保存し、このデータをデータベースに格納する。

要素は直接参照する。
[0106] 属性:

- ・targetElement:必須。SMLからvalueの内容を割り当てる要素(W3C SMIL2.0と同様)。

【0107】・targetAttribute:任意選択。SMLからvalueの内容を割り当てるターゲット属性の属性(SMIL2.0のattribute

「value」属性と同等)。指定しない場合は、「value」になる。

【0108】・test:任意選択。認識結果を割り当て
る際の条件を指示するXML Pattern (W3C
XMLDOM仕様と同様) ストリング。デフォルト

条件は真。
【0109】・value:必須。ターゲット要素に割
り出せる稼働率ドキュメントの値を指定するXPAT

【0110】例：上記のSMLのリターンを与えられる

と、以下の `reco` 要素は `bydest_city` 中の値を、
`target_box` ページの要素 `txtBoxOrigin` におよ

びtxtBoxDesに転送する。

ベルブラウザのみ。
【0125】2. 3. 1 Start
Startメソッドは、明示的には非活動化してはいない
認識コンテキストについてのすべての最上位規則をアク
ティブな文法として使用して認識プロセスを開始する。
【0126】構文: Object. Start ()
戻り値: なし
例外: なし
例外: なし
例として、顧客が非ゼロのステータスコードをセ
ットし、顧客があった際はonNoRecoイベントを
発生させる。可能性のある障害には、文法が存在しない
(recoステータス=1)、文法のコンパイルの失
敗、存在しないURIなど様々な原因になりうる文法の
ロードの失敗 (recoステータス=2)、あるいは
音声プラットフォームのエラー (recoステータス=
=3) などが含まれる。

【0127】2. 3. 2 Stop
Stopメソッドは、認識プロセスを終了する呼び出し
である。Recoオブジェクトはオーディオの記録を中
止し、レコグナイザは、記録が中止される時点までに受
け取ったオーディオについての認識結果を返す。Rec
oが使用するすべての認識リソースは解放され、その文
法は非活動化される。(このメソッドは、自動モードに
よる通常の認識には明示的に使用する必要があることに
留意されたい。これは、レコグナイザ自体が、完全な文
法を認識した後のエンドポイント検出においてrecoオ
ブジェクトを中止するからである。) Recoが開始さ
れていない場合、この呼び出しは効果を持たない。
【0128】構文: Object. Stop ()
戻り値: なし
例外: なし

【0129】2. 3. 3 Cancel
Cancelメソッドは、レコグナイザへのオーディオ
の供給を中止し、文法を非活動化し、レコグナイザを解
放し、すべての認識結果を放棄する。ブラウザは、キャ
ンセルされた認識についての認識結果は放棄する。レコ
グナイザが解放されていない場合、この呼び出しは効果
を持たない。
【0130】構文: Object. Cancel ()
戻り値: なし
例外: なし

* 40

インラインHTML	<Reco onReco="handler">
イベントプロパティ	Object.onReco = handler;
	Object.onReco =
	GetRef("handler");

【0138】イベントオブジェクト情報:
【0139】

【表2】

はデータについてイベントオブジェクトに照会を行うこ
とができる。
【0148】2. 4. 3 onTimeout
onTimeoutは、通例は音声プラットフォームか
らのエラーを反映する2タイプのイベントを扱う。
【0149】・認識が完了する前にmaxTime属性
で指定された期間を過ぎた (2. 2. 1参照) ことを通
知する、タグインタープリタが投入するイベントを扱
う。このイベントは通例、分岐型アーキテクチャで生じ
うる問題を反映する。

* 10

インラインHTML	<Reco onTimeout="handler">
イベントプロパティ	Object.onTimeout = handler
(ECMAScript)	Object.onTimeout =
	GetRef("handler");

【0154】イベントオブジェクト情報:
【0155】

※ 【表6】

バブル	なし
転送するには	認識の中止前に、maxTime 属性で指定された期間が過ぎ るとブラウザが投入する
デフォルトアクション	reco ステータスを-12にセットする

【0156】イベントプロパティ: イベントハンドラは
プロパティを直接受け取ることはないが、ハンドラはデ
ータについてこのイベントオブジェクトに照会を行うこ
とができる。
【0157】2. 4. 4 onNoReco: onNo
Recoは、有効な認識結果を返すことができない際に
音声認識プラットフォームが投入するイベント用のハン★

インラインHTML	<Reco onNoReco="handler">
イベントプロパティ	Object.onNoReco = handler;
	Object.onNoReco =
	GetRef("handler");

【0160】イベントオブジェクト情報:
【0161】

【表8】

* 【0150】・また、(i) 認識が開始されたが、b
abbleTimeoutで指定された期間内に認識が
ないまま処理が中止した際に、音声認識プラットフォームが投入するイベントも扱う (2. 2. 1参照)。
【0151】このイベントは、認識プロセスを自動的に
キャンセルする。
【0152】構文:
【0153】
【表5】

・ `element` の必要な属性を指定することができる。これは、例えば、HTML フォームコントロール中の値をデリファレンスするのに有用である。下の例で、`textBoxOrigin` 要素および `textBoxDest` 要素の `value` 属性を、プロンプトの出力前にテキストに代入している。

```
(prompt id="confirm")
    あなたが行きたいのは
    (value targetBleucant="textBoxOrigin"
    targetAttribute="value" /)
    から
    (value targetBleucant="textBoxDest"
    targetAttribute="value" /)
    ですか?
    (prompt)
```

[0172] 3. 1. 3 オーディオファイル *
(prompt)
ピーツという音がしたらメッセージを録音してください
(value href="/wav/deep.wav" /)
(prompt)

[0173] 3. 1. 4 参照プロンプト
インラインの内容を指定する代わりに、`src` 属性を空要素とともに使用し、URI を介して外部の内容を参照することができる。例えば、
(prompt id="welcome"
src="/ACQRefatherPrompts/welcome" /)
`src` 属性の対象は、インラインプロンプトに指定する上記の内容の任意部分またはすべてを保持することができる。

[0174] 3. 2 属性およびプロパティ
このプロンプト要素は、以下の属性 (ダウンレベルブラウザ) およびプロパティ (ダウンレベルおよびアップレベルブラウザ) を保持する。
[0175] 3. 2. 1 属性
・ `atts`: 任意選択。テキストから音声への合成用のマークアップ言語タイプ。デフォルトは「SAPI 5」。
[0176] ・ `src`: インラインプロンプトを指定する場合に任意選択。参照するプロンプトの URI (3. 1. 4 参照)。

[0177] ・ `bargein`: 任意選択。整整。プロンプトの開始から、人間の聴者が再生を中断できるようなまでのミリ秒単位の時間。デフォルトは無限。すなわちバーgeinを許可しない。 `bargein=0` にすると、即時のバーgeinが可能になる。これは、ブラウザフォアグラウンドする時間のバーgeinにも該当する。 `recog` を開始時にどちらを使用可能にするかに応じて、キーワードまたはエネルギーベースのバーgein時間とこの方式で構成することができる。

```
//
</script>
<script for="window" event="onload">
<!--
    news.Start 0; // keyword.Start 0;
//
</script>
</head>
<body>
```

<prompt id="news" bargein="0">

水曜日の株式市場も、投資家が、来週の連邦準備理事会の会合に先立ち大きな動きにつながる材料を得られなかったことから底限に活気がありませんでした。ハイテク銘柄中心のナスダック総合指数は42.51ポイント下落し、2156.26で取引を終えました。ダウジョーンズ工業平均株価は、午後に入って反騰がなく17.05ポイント下落して10866.46で取引を終えました。

※プロンプトDOMオブジェクトは以下のイベントをサポートするが、そのハンドラはプロンプト要素の属性として指定することができる。
[0193] 3. 4. 1 onBookmark
合成ブックマークに遭遇すると発生する。このイベントは再生を一時停止しない。
[0194] 構文:
[0195]
[表9]

20

```
(prompt)
    (recog id="keyword")
    reject="70"
    onReco="checkWbBargein()"
    (grammar
    src=http://denail/news bargein grammar.xml /)
    (recog)
    (body)
    (html)
```

[0192] 3. 4 プロンプトイベント *

インラインHTML	SpeechOnBookmark="handler">
イベントプロパティ	ObjectOnBookmark="handler"
	ObjectOnBookmark="GetRef(handler)"

※ [表10]

[0196] イベントオブジェクト情報:

[0197]

バブル	なし
起動するには	レンダリングしたストリング中のブックマークに遭遇する
デフォルトアクション	ブックマークストリングを戻す

[0198] イベントプロパティ: イベントハンドラは イベントを構成するかの決定はプラットフォームによるプロパティを直接受け取ることはないが、ハンドラはデータについてこのイベントオブジェクトに照会を行うことができる。

[0199] 3. 4. 2 onBargein: ユーザのバーgeinイベントを抽出すると発生する。(例えばエネルギー検出やキーワード認識など、何がバーgein

イベントを構成するかの決定はプラットフォームによることに留意されたい。) このイベントハンドラを指定しても、自動的にバーgein機能がオンになるわけではない。
[0200] 構文:
[0201]
[表11]

```

]]</script>
<body>
  <input name="textBoxOrigin" value="Seattle"
    type="text"/>
  <input name="textBoxDest" type="text" />
  ...
  <prompt id="confirm" onBargein="interrupt0"
    bargein="0">
    <bookmark mark="mark_origin_city" />
    <value targetElement="origin"
      targetAttribute="value" />から
    <bookmark mark="mark_dest_city"
      />行きたい先地を書いて下さい
    </prompt>
    <reco onReco="ProcessCityConfirm0">
      <grammar src="/grm/1033/cities.xml" />
    </reco>
    ...
  </body>
</DTMF>
例1：テキストにキーをマッピングする
<input type="text" name="city" />
<DTMF id="city_choice" timeout="2000"
  numDigits="1">
  <dtmfgrammar>
    (key values="1")シアトル(key)
    (key values="2")ホuston(key)
  </dtmfgrammar>
  <bind targetElement="city"
    largeAttribute="value" />
  </DTMF>
city__choice) を起動して、ユーザが1を
押すと「Seattle」が入力フィールドに割り当て
られ、2を押すと「Houston」が割り当てられ、そ
の他の場合は何も割り当てられない。
【0216】例2：どのようにしてDTMFを複数フィ
ールドに使用することができるか
<input type="text" name="area_code" />
<input type="text" name="phone_number" />
<DTMF id="area_code" numDigits="3">
  onReco="extension.Activate()"
  <bind targetElement="area_code" />
</DTMF>
<DTMF id="extension" numDigits="7">
  <bind targetElement="phone_number" />
</DTMF>
この例は、いかにしてユーザが複数フィールドに入力す
るのを可能にするかを示している。
【0217】例3：音声入力およびDTMF入力をとら
50 に許可し、ユーザがDTMFを開始した際に音声を使用
```

4. 4. 1 onkeypress
DTMFキーを押すと発生する。これは、HTMLコン
トロールから継承したデフォルトイベントを上書きす
る。ユーザがエスケープキーを押すと、onKeyPress
*essではなくonRecイベントが発生する。
【0231】構文：
【0232】
【表15】

インラインHTML	<DTMF onkeypress="handler">
イベントプロパティ	Object.onkeypress = handler
	Object.onkeypress =
	GetRef("handler");

【0233】イベントオブジェクト情報：
【0234】※ 10 ※ 【表16】

バブル	なし
起動するには	タクトーン電話のキーパッドを押す
デフォルトアクション	押されているキーを戻す

【0235】イベントプロパティ：イベントハンドラは
プロパティを直接受け取ることはないが、ハンドラはデ
ータについてこのイベントオブジェクトに照会を行うこ
とができる。
【0236】4. 4. 2 onReco 20 【表17】
DTMFセッションを終了すると発生する。このイベン★

インラインHTML	<DTMF onReco="handler">
イベントプロパティ	Object.onReco = handler
	Object.onReco =
	GetRef("handler");

【0239】イベントオブジェクト情報：
【0240】☆ 【表18】

バブル	なし
起動するには	ユーザがエスケープキーを押す、またはキーストローク の回線が指定の直を渡す
デフォルトアクション	押されているキーを戻す

【0241】イベントプロパティ：イベントハンドラは
プロパティを直接受け取ることはないが、ハンドラはデ
ータについてこのイベントオブジェクトに照会を行うこ
とができる。
【0242】4. 4. 3 onTimeout 20 【表19】
タイムアウトまでに、句の終了イベントを受け取らない◆

インラインHTML	<DTMF onTimeout="handler">
イベントプロパティ	Object.onTimeout = handler
	Object.onTimeout =
	GetRef("handler");

【0245】イベントオブジェクト情報：
【0246】【表20】


```

        <bind targetElement="textBoxOrigin"
        value="//origin_city" />
        <bind targetElement="textBoxDest"
        test="/sal/dest_city[@confidence >= 40]"
        value="//dest_city" />
    </sp:recog>
</t:par>
</body>
</html>

```

【0268】7. SMEX (メッセージ) 要素/オブジェクト 10 が非同期的な、アプリケーション開発者がタイムアウト設定を操作するための内蔵クロックも有する。

【0270】メッセージまたは smex オブジェクトは、通信手段によってアノノスティック (anonymous) である。しかし、一実施形態では、smex オブジェクトは、通常の XML やマークアップ要素と同じ寿命を有する。すなわち、smex オブジェクトは、それをホストするドキュメントをアンロードすると消滅する。多くのケースでは、smex オブジェクトはアンロードされると自動クリーンアップを実行し、通信リソースを解放することができるが、マークアップページ間で永続的な通信リンクが望ましい使用事例 (例えば呼の制御など) もありうる。そのような事例のために、このアプリケーションチャチャでは、割り振られたリソースを解放する (例えばソケットを閉じるなど) 責任をアプリケーション開発者に課す。

【0271】smex オブジェクトは、メッセージのフォーマット (スキーマ) についてはニュートラルである。実施形態によっては、既存の標準的なメッセージフォーマット (例えば SOAP または CCXML で使用するスキーマをサポートすることを必要とするのが望ましい場合もある。基本的には、このアプリケーションチャチャは、プラットフォーム開発者およびアプリケーション開発者の両者が、XML あるいはそれに類似のマークアップの規格化された拡張性を最大限に活用して、一方では相互操作性を失うことなく他の機能を導入することを可能にする。

【0272】

例 1 : ロギングオブジェクトとしての smex の使用

```

<smex_id="logServer">
  <param name="d:server"
  xmlns:d="urn:microsoft.com/COM">
    <d:protocol>DCOM</d:protocol>
    <d:clsid>709309302020202020942098432098</d:clsid>
    <d:iid>909309302020202020942098432098</d:iid>
  </param>
  </smex>
</listen...>
...//recog結果を入力フィールドにバインドする他のディレクティブ
..... <bind targetElement="logServer"

```

ッセージは、次の onReceive イベントが送られるのできる状態になるまで、value として使用することができる。

【0281】timer : 読み取り/書き込み。タイムアウトイベントをトリガするまでの時間を表すミリ秒単位の数。クロックは、このプロパティに正の値が割り当てられると刻時を開始する。この値は、カウントダウンの進行中に変更することができる。ゼロまたは負の値にすると、タイムアウトイベントをトリガせずにクロックを停止する。デフォルトは 0、すなわちタイムアウトなしである。

【0282】status : 読み取り専用。オブジェクトの最近のステータスを表す整数。可能な値は、0、-1、および -2 であり、それぞれ、正常、タイムアウトの終了、およびプラットフォームとの通信を確立できない、あるいは通信の中断を意味する。受信されるプロパティを通じて、プラットフォーム固有のエラーメッセージを伝達するとよい。エラーメッセージの伝達が成功した場合、ステータスコードは 0 になる。

【0283】7. 2 イベント
このオブジェクトは以下のイベントを有する。

【0284】onReceive : このイベントは、プラットフォームメッセージが到着すると送られる。バインド要素によって宣言されたディレクティブがある場合には、このイベントを発生させる前にそのディレクティブを先に評価する。イベントを送る前に、受け取ったプロパティを更新する。

【0285】onError : このイベントは、タイムアウトが経過したとき、あるいは通信リンクエラーに遭遇したときに送られる。このイベントを送る際、上記のように、ステータスプロパティをそれに対応するエラー *

```

<smex id="logServer" onload="addFunction()">

```

```

</smex>

```

```

<script>

```

```

function my_logMessage(logClass, message) {
  logServer.send = logClass + "|" + message;
}

```

```

function addFunction() {
  logServer.prototype.logMessage=

```

```

my_logMessage;
}

```

```

</script>

```

よりオブジェクト指向的な方式でこの情報を参照することができる。

```

logServer.logMessage(RECOG_ERROR, "My message");

```

```

}

```

上記の例のように拡張を機能させるために、smex オブジェクトの実装者にはより多くの作業が要求されるが、すべての必要な機能はすでに確立された規格であることに留意されたい。

【0289】

【発明の効果】以上、説明したように、本発明によれば

【図面の簡単な説明】
【図1】本発明の実施形態の、コンピュータ用デバ

ある装置の形を仮定するとき、smex は以下の主要要素を有することができる。
・bind : ディレクティブを受信メッセージに作用させる点を除いては、reco の場合と同様。
・param : reco の場合と同様。smex オブジェクトのプラットフォーム固有のパラメータを提供する。各 param 要素は、「name」属性を使用して名前をつけることができ、param 要素の内容がそのパラメータの値になる。一実施形態では、この要素は、ネームスペースの構造的な XML 属性と XML データタイプ宣言を理解しているべきである。
【0287】7. 4 その他の補足説明
ロギング機能のために SMEX を拡張する簡潔な方法の一つが以下である。
(smex id="logServer"...) ... </smex>
<script> function logMessage(logClass, message) { logServer.send = logClass + "|" + message; } </script>

これは、実際に、その振る舞いを個別に設定することのできる (グローバル) 関数でこのオブジェクトを拡張している。上の例では、ID とメッセージの間にフィールド区切り文字「|」を挿入するようにロギング関数をプログラムしている。
【0288】グローバル関数を好まない者は、ECMA Script の「prototype」プロパティを使用して、この関数をオブジェクトメソッドとして付加することができる。例えば、

【図18】

```
<reco id="reco_cream_sugar"><grammar src="/cream/sugar"/>
  <bind test="/@confidence >= 10 and
    host()<br>get_drink/drink = 'coffee'"/>
    targetElement="cream" targetAttribute="checked"
    value="/cream/value"
    targetElement="sugar" targetAttribute="checked"
    value="/sugar/value"
    targetElement="confirm" targetMethod="start"
    targetElement="reco_yesno" targetMethod="start"/>
</reco>

<reco id="reco_yesno"><grammar src="/yesno"/>
  <bind test="/yes[@confidence >= 10]/>
    356 targetElement="thanks" targetMethod="start"
    358 targetElement="get_drink"
    targetMethod="submit"/>
  <bind test="/no or /@confidence < 10"/>
    targetElement="retry" targetMethod="start"
    targetElement="ask" targetMethod="start"
    targetElement="reco_drink"
    360 targetMethod="start"/>
</reco>

<!-- call control section -->
<smx id="telephone" sent="start_listening"><param
  server="ccxmlproc"> - </param>
  <bind targetElement="uid" value="/guid"/>
  <bind test="/Call_connected"
    361 targetElement="welcome" targetMethod="start"
    362 targetElement="ask" targetMethod="start"
    363 targetElement="reco_drink"
    targetMethod="start"/>
  </smx>
</body>
</html>
```

フロントページの続き

(51) Int. Cl. ⁷	識別記号	F I	5 7 1 J	7-23-D (参考)
(72) 発明者	ワアン クアンサン	(72) 発明者	ホン ショウウエン	
	アメリカ合衆国 98006 ワシントン州		アメリカ合衆国 98006 ワシントン州	
	ベルビュー サウスイースト 48 コート		ベルビュー サウスイースト 58 プレイ	
	16470		ス 17797	
			ドターム(参考)	50015 KK00 LL11

【図13】

```
ASPページの一例
<!-- Page language="JavaScript" AutoEventWireup="false" Inherits="Credit.Transaction" -->
<html><head>
  <!-- ASPX page for both voice-only & multimodal credit card example -->
</head>
<script>
function handle() {
  if (field == get_card_info.card_num) {
    if (get_card_info.card_num.length != 15) {
      prompt.speak ("visa should have 15 digits");
      get_card_info.card_num = "";
    } else {
      if (get_card_info.card_num.length != 16) {
        prompt.speak ("visa should have 16 digits");
        get_card_info.card_num = "";
      }
    }
  }
  if (get_card_info.card_type == "credit") {
    str = "card type: " + get_card_info.card_type;
    str += "card number: " + str + "<br>";
    str += "card number: " + str + "<br>";
    str += "expiry date: " + str + "<br>";
    return str;
  }
}

</script>
<script runat="server">
function Page_Load (obj, args) {
  if (PostBack) {
    Validator = new System.Web.UI.WebControls.CreditCardValidator ("~/CreditCardValidator.aspx");
    Validator.LocateRequiredControl (args);
    Navigate (ChoosePage (dml));
  } else {
    // initialize fields with args
  }
}
</script>
</head>
<body>
<speech:form id="get_card_info" style="system_initiative"
  prompt="~/prompt/get_payment" onsubmit="handle()"
  grammar="~/grammar/card_type" prompt="What credit card would you use?"
  <option>American Express</option>
  <option>MasterCard</option>
  </speech:form>
  <speech:form name="card_number" prompt="Please say the number"
  grammar="~/grammar/digit" onphrasefinish="handle()"
  <speech:form name="expiry_date" prompt="What is the expiration date?"
  grammar="~/grammar/date" onphrasefinish="handle()"
  </speech:form>
</body>
</html>
```

前記レコグナイズはジェスチャレコグナイズを含み、前記文法はジェスチャ認識に関連することを特徴とする請求項1に記載のコンピュータ読み取り可能な記録媒体。

【請求項7】

前記レコグナイズは視覚レコグナイズを含み、前記文法は視覚認識に関連することを特徴とする請求項1に記載のコンピュータ読み取り可能な記録媒体。

【請求項8】

クライアント/サーバネットワークにおける音声認識の方法であって、入力音声を表すデータと、認識を行うために、前記入力を表すデータに使用する文法の指示とを、ネットワークを介して受信する受信ステップと、レコグナイズとともに前記文法を使用して前記データを処理し、認識結果を得る処理ステップと、

前記入力を表すデータについての前記認識結果を、前記ネットワーク上の遠隔位置に送信する送信ステップと

を備えたことを特徴とする方法。

【請求項9】

前記指示は、前記文法の位置へのリファレンスを提供することを特徴とする請求項8に記載の方法。

【請求項10】

前記指示は、認識用の言語へのリファレンスを含むことを特徴とする請求項8に記載の方法。

【請求項11】

前記遠隔位置にプロンプトを提供する提供ステップをさらに備えたことを特徴とする請求項8に記載の方法。

【請求項12】

プロンプトを提供する提供ステップは、テキストデータを音声データに変換するステップと、該音声データを前記遠隔位置に提供するステップとを含むことを特徴とする請求項11に記載の方法。

【請求項13】

クライアント/サーバシステム中のクライアントデバイスで実行するためのマークアップ言語を有するコンピュータ読み取り可能な記録媒体であって、

前記マークアップ言語は、非表示式、音声入力ベースのクライアントデバイス、およびマルチモーダルベースのクライアントにおける、認識関連イベント、GUIイベント、および電話イベントのうち少なくとも1つを、前記クライアントデバイスの各々と対話するウェブサーバのために、統一する命令を備えたことを特徴とするコンピュータ読み取り可能な記録媒体。

【請求項14】

前記マークアップ言語は、HTML、XML、cHTML、XML、およびWMLの1つを含むことを特徴とする請求項13に記載のコンピュータ読み取り可能な記録媒体。

【請求項15】

前記マークアップ言語は、スクリプティング言語を含むことを特徴とする請求項13に記載のコンピュータ読み取り可能な記録媒体。

【請求項16】

前記マークアップ言語は、同期化マルチメディアマークアップ言語を含むことを特徴とする請求項13に記載のコンピュータ読み取り可能な記録媒体。

【請求項17】

前記マークアップ言語はスクリプティングを模倣することを特徴とする請求項13に記載の方法。